CS 201 Group Project
Text-Based Calculator

Luke Underwood
Andrew Ng
Stefano Fochesatto

October 8, 2019

Source Code Link: `https://bit.ly/2nvsHe7`
This project took approximately XX hours to complete.

1 Pitch

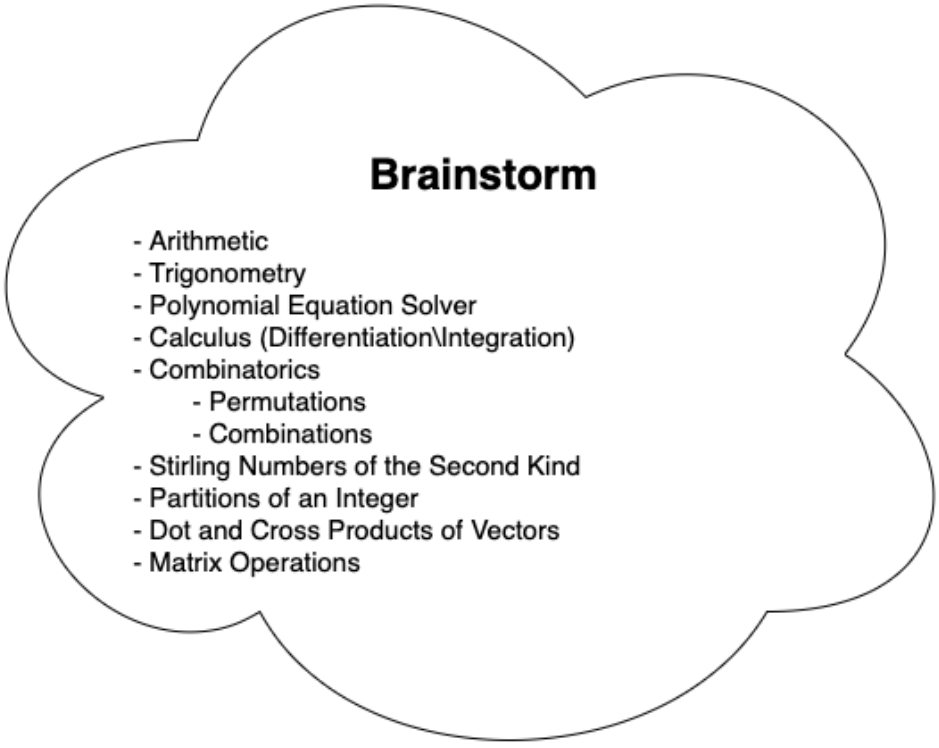
For our project we want to create a text-based calculator. Some of the features we want to implement beyond standard arithmetic are trigonometric functions, differentiating and integrating with single or multi-variables, polynomial equation solver, various combinatorial functions ie. Stirling Numbers, Bell Numbers and Partitions of an integer.

2 Design

2.1 Overall Design

With our calculator we want to take a more subdued approach. Our goal is to create a no nonsense, no frills calculator that provides some of the same features that are only present in industry standards like, MATLAB and Mathmatica. The inspiration for

our project comes from the fact that our group consists entirely of STEM majors, so it's important to us to be able to do a number of calculations accurately and efficiently. What makes our approach unique is that we want to create a calculator doesn't have the steep learning curve that other programs have.



2.2 Prior Art

The idea of a calculator has been fairly well explored since Charles Babbage's mechanical computers (`https://en.wikipedia.org/wiki/Charles_Babbage`) and even going back to the abacus. However, out project will likely attempt to replicate such calculators as the default windows calculator, while integrating features such as the

one built into google that allows the user to simply type in basic arithmetic. In addition, we would like to integrate some features that are seen in wolfram alpha's mathmatica, while making them more accessible to the average user.

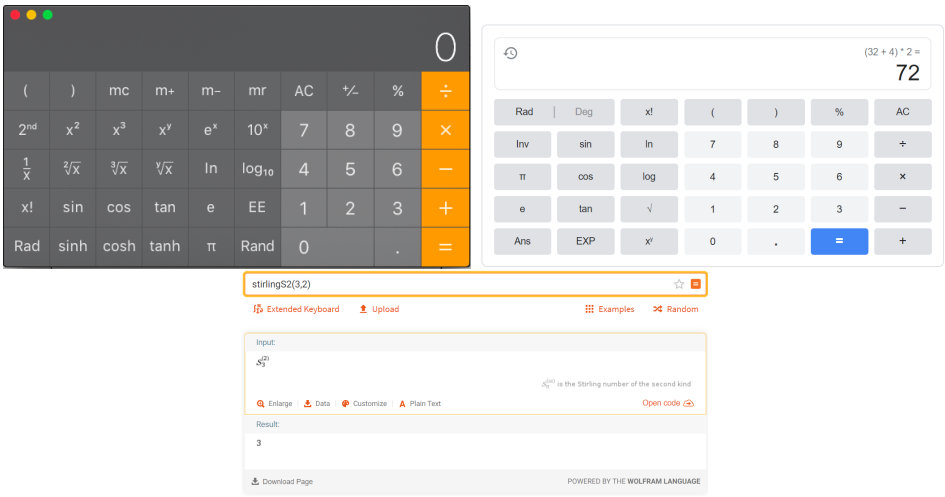


Figure 1: Sample Calculators

2.3 Technical Design

For how the program will look like to the user, it will mostly be menu-based on the command line interface, which we will implement using switch statements. The main function will likely be where we will put our menu prompts, and it will then call math operations that all have their own functions defined in a separate header and source file. We also want a line-based evaluation feature, where we will have a function that takes in a string stream and parses the performed operations. This will be limited to basic arithmetic, and possibly trig functions, exponents, and logarithms.

2.4 Required Libraries

For this application, we will likely use libraries such as iostream, vector, string, and sstream, for basic input, output, and data handling. In addition, we will likely use cmath to handle any computations that go beyond basic arithmetic. For anything beyond the capabilities of cmath, we will likely implement other libraries such as The GNU multiple precision arithmetic library (GMP), or others that will make the handling of large numbers and high level math simpler.

3 Project Update

In this section, each group member needs to give an update on the work they are doing.

3.1 Group Member 1 Name

In this subsection, write about your contribution to the group project. Include three artifacts of your work. Artifacts (like the one in Figure 2 can be:

- Charts of data
- Screenshots of your work
- Screenshots showing tool usage

3.2 Group Member 2 Name

In this subsection, write about your contribution to the group project. Include three artifacts of your work. Artifacts (like the one in Figure 3 can be:

- Charts of data
- Screenshots of your work
- Screenshots showing tool usage

Figure 2: GROUPMEMBERNAME1 Artifacts

4 Results

A screenshot of our program is displayed in Figure 4. In this section, talk about the results of your program.

5 Post Mortem

In this section, you will write a paragraph in about 100 words about what went right and what went wrong for your implemen-

Figure 3: GROUPMEMBERNAME2 Artifacts

tation. What lessons were learned or best practices identified?

6 Sample Output

In this section, write about the sample output of your program. For example, Figure 4 is a screenshot of a ROBOT parameter visualization screen in our program.

7 Program Source Code

The source code is licensed under the WRITETHENAMEOFYOURLICENSEHERE license.

7.1 Triangle Header

```
#ifndef TRIANGLE_HPP
#define TRIANGLE_HPP

class Triangle
{
public:
    Triangle();
    ~Triangle();

    void print();
}

#endif
```

7.2 Triangle Source

```
#include <iostream>
#include "triangle.hpp"

Triangle::Triangle()
{
    // Constructor
}

Triangle::~Triangle()
{
    // Deconstructor
}

Triangle::print()
{
    std::cout << "Triangle!" << std::endl;
}
```

References

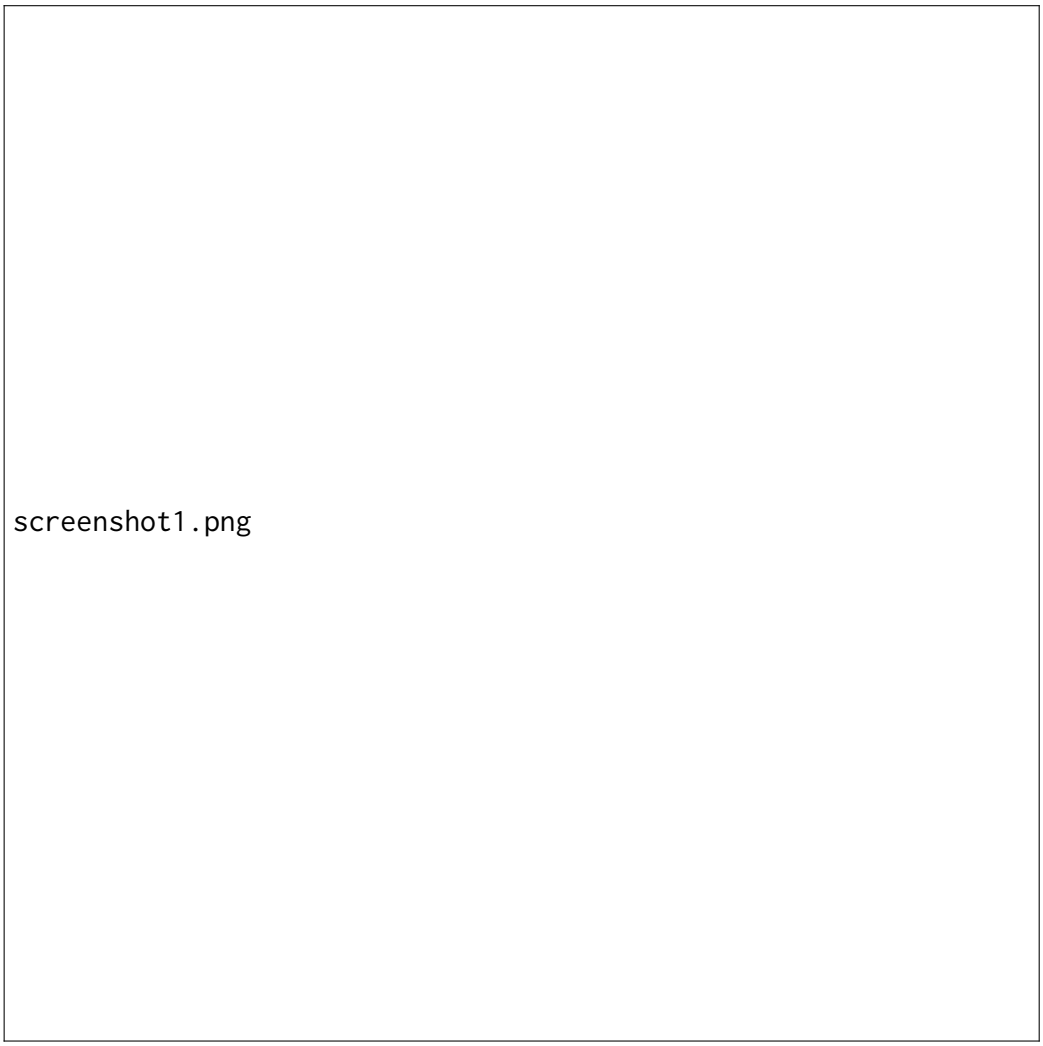


Figure 4: A Screenshot of our finished application.